UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/760,031 | 01/12/2001 | Robert H. Halstead JR. | 09612.1014-02000 | 1846 |

22852 7590 12/21/2006

FINNEGAN, HENDERSON, FARABOW, GARRETT & DUNNER LLP
901 NEW YORK AVENUE, NW
WASHINGTON, DC 20001-4413

| EXAMINER | |
|---|---|
| KANG, INSUN | |
| **ART UNIT** | **PAPER NUMBER** |
| 2193 | |

| SHORTENED STATUTORY PERIOD OF RESPONSE | MAIL DATE | DELIVERY MODE |
|---|---|---|
| 3 MONTHS | 12/21/2006 | PAPER |

**Please find below and/or attached an Office communication concerning this application or proceeding.**

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

| Office Action Summary | Application No. | Applicant(s) | |
|---|---|---|---|
| | 09/760,031 | HALSTEAD ET AL. | |
| | Examiner | Art Unit | |
| | Insun Kang | 2193 | |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *25 September 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *1-24 and 26-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-24 and 26-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

## DETAILED ACTION

1. This action is in response to the amendment filed 9/25/2006.

2. As per applicant's request, claims 1, 11, 21, 22, and 26 have been amended. Claims 1-24 and 26-30 are pending in the application.

### *Double Patenting*

3. The applicant asked the Examiner to hold the rejection; therefore, the rejection under double patenting is maintained. See the previous action.

### *Claim Rejections - 35 USC § 103*

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1, 11, 21, and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Linked list code examples from *Data Structures and other Objects Using C++* by Main and Savitch (1997) listed in the cs.appstate.edu website (Index of...examples; "Bag Implementation Using Linked Lists," 1998) hereinafter "Main" in view of Blainey (US Patent 6,045,585).

As per claim 1, Main discloses defining an object with defined fields to support values in preallocated memory space and with an option data structure which supports references to option values preallocation of memory space for the full option values; accessing a field value stored in one of the defined fields and accessing an option value not stored in the defined fields in the object using expressions of the same syntactic form (i.e. see the Bag class). Main does not explicitly discloses an actual compilation process, during compilation, determining whether at least one of the expressions accesses one of (a) a field value or (b) an option value;when it is determined that the expression accesses a field value, compiling the expression into a first code for accessinq the field value; and when it is determined that the expression accesses an option value, compiling the expression into a second code for accessing the option value. However, when code is compiled, the compiler performs a data type check. Specifically Blainey teaches that data type check performed by a compiler was known in the pertinent art, at the time applicant's invention was made, so that information stored there can be determined. It would have been obvious for one having ordinary skill in the art to modify Main's disclosed system to incorporate the teachings of Blainey (i.e. col. 2 lines 27-40). The modification would be obvious because one having ordinary skill in the art would be motivated to determine data type for appropriate memory allocation based on the data type.

As per claims 11 and 21, they are the system versions of claim 1, respectively, and are rejected for the same reasons set forth in connection with the rejection of claim 1 above.

As per claim 22, it is the product version of claim 1 respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 1 above.

6. Claims 1-8, 11-18, 21, 22-24, and 26-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over McLennan ("Object-Oriented Programming with [incr Tcl] Building Mega-Widgets with [incr Tk]," 1996), in view of Linked list code examples from *Data Structures and other Objects Using C++* by Main and Savitch (1997) listed in the cs.appstate.edu website (Index of...examples; "Bag Implementation Using Linked Lists," 1998) hereinafter "Main," and further in view of Blainey (US Patent 6,045,585).

As per claim 1, McLennan discloses defining an object with defined fields to support values in preallocated memory space and with an option data structure which supports references to option values (i.e. page 87 lines 24-35). McLennan does not explicitly teach defining the option values without preallocation of memory space for the full option values. However, Main teaches such dynamic allocation was known in the pertinent art, at the time applicant's invention was made, to save memory space such as the Bag implementation with linked list disclosed in Main. It would have been obvious for one having ordinary skill in the art to modify McLennan's disclosed system to incorporate the teachings of Main. The modification would be obvious because one having ordinary skill in the art would be motivated to use dynamic memory that shrinks and grows as needed for efficient memory space usage as suggested by Main.

McLennan in view of Main further discloses accessing a field value stored in one of the defined fields and accessing an option value not stored in the defined fields in the object using expressions of the same syntactic form (i.e. page 87).

McLennan and Main do not explicitly disclose during compilation, determining whether at least one of the expressions accesses one of (a) a field value or (b) an option value; when it is determined that the expression accesses a field value, compiling the expression into a first code for accessinq the field value; and when it is determined that the expression accesses an option value, compiling the expression into a second code for accessing the option value. However, when code is compiled, the compiler performs a data type check. Specifically Blainey teaches that data type check performed by a compiler was known in the pertinent art, at the time applicant's invention was made, so that information stored there can be determined. It would have been obvious for one having ordinary skill in the art to modify McLennan in view Main's disclosed system to incorporate the teachings of Blainey (i.e. col. 2 lines 27-40). The modification would be obvious because one having ordinary skill in the art would be motivated to determine data type for appropriate memory allocation based on the data type.

As per claim 2, McLennan discloses a method as claimed in claim 1 wherein the option data structure identifies change handler code that is executed when an option value changes (i.e. page 81 lines 1-3).

As per claim 3, the rejection of claim 2 is incorporated and McLennan further discloses a method as claimed in claim 2 wherein change handler code for one option is

defined in different classes within a class inheritance hierarchy and the change handler code from each class is executed when the option value changes (i.e. page 81 lines 1-3).

As per claim 4, the rejection of claim 1 is incorporated and McLennan further discloses a method as claimed in claim 1 wherein the option data structure includes a type description of the option value, the method further comprising: during compilation, using the type description in the option data structure to process an operation on the option value (i.e. page 79 lines 3-9).

As per claim 5, the rejection of claim 1 is incorporated and McLennan further discloses a method as claimed in claim 1 wherein an option data structure includes a default value, the method further comprising, in a get operation to an instance of the class, if an option value which applies to the instance has been set, getting the set option value and, if no value which applies has been set, getting the default value for the class (i.e. page 79 lines 3-9).

As per claim 6, the rejection of claim 1 is incorporated and McLennan further discloses a method as claimed in claim 1 defining a first class with a first option data structure of a first form which supports, in instances of the class, references to option values without preallocation of memory space for the full option values; defining a second class with a second option data structure of a second form which supports, in instances of the second class, references to option values without preallocation of memory space for the

full option values, the second form being different from the first form; and during compilation, encoding an option operation as a method call to an object of the first class and to an object of the second class without regard to the form of the option data structure supported by the class (i.e. page 79).

As per claim 7, McLennan discloses a method as claimed in claim 1 further comprising: notifying objects of a change in an option value through a change handler identified by an option binding, the option binding being located by first searching a mapping data structure for a previously computed mapping to the option binding and, if no mapping was previously computed, by then computing the mapping to the option binding and storing the mapping in the mapping data structure (i.e. page 81 lines 1-3). As per claim 8, the rejection of claim 1 is incorporated and Main further discloses a method as claimed in claim 1 wherein the option data structure comprises a linked list of option items having option values (Bag implementation with Linked list).

As per claims 11-18, they are the system versions of claims 1-8, respectively, and are rejected for the same reasons set forth in connection with the rejection of claims 1-8 above.

As per claim 21, it is the data processing system version of claim 1 respectively, and is rejected for the same reasons set forth in connection with the rejection of claim 1 above.

As per claims 22-24, they are the product versions of claims 1, 7, and 8, respectively, and are rejected for the same reasons set forth in connection with the rejection of claims 1, 7, and 8 above.

As per claims 26-30, they are the method versions of claims 1-6 respectively, and are rejected for the same reasons set forth in connection with the rejection of claims 1-6 above.

7. Claims 9, 10, 19, and 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over McLennan ("Object-Oriented Programming with [incr Tcl] Building Mega-Widgets with [incr Tk]," 1996), in view of Linked list code examples from *Data Structures and other Objects Using C++* by Main and Savitch (1997) listed in the cs.appstate.edu website (Index of...examples; "Bag Implementation Using Linked Lists," 1998), hereinafter "Main," further in view of Blainey (US Patent 6,045,585). and still further in view of Hostetter et al ("Curl: A Gentle Slope Language for the Web," World Wide Web Journal, spring, 1997, art of record) hereinafter "Hostetter."

As per claim 9, McLennan, Main, and Blainey do not explicitly disclose a nonlocal option value applies to other objects in a nonlocal option hierarchy. However, Hostetter teaches a nonlocal option value applies to other objects in a nonlocal option hierarchy (see Section3, Page 4, Lines 1-2, "The screen shot above reflects the fact the user has selected something besides the default color (red) and quantity (0)."). Color is a nonlocal option because all text in a given document is usually the same color. Therefore, it would have been obvious to one of ordinary skill in the art at the time the

invention was made to incorporate the teaching of Hostetter into the method of McLennan, to comprise a nonlocal option value that applies to other objects in a nonlocal option hierarchy. The modification would have been obvious because one of ordinary skill in the art would have been motivated to implement properties in a dynamically bound environment using a deep binding mechanism.

As per claim 10, McLennan, Main, and Blainey do not explicitly disclose that the nonlocal option hierarchy is a graphical hierarchy. However, Hostetter teaches that the nonlocal option hierarchy is a graphical hierarchy *(see* Section3, Page 4, Lines 12, "The screen shot above reflects the fact the user has selected something besides the default color (red) and quantity (0);" Section 4.3, Page 9, Lines 34-35, "text. Properties control the color, size and font family as well as indicating whether the text should be bold or italic."). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Hostetter into the method of McLennan. The modification would have been obvious because one of ordinary skill in the art would have been motivated to represent a graphic image as a hierarchical tree of Graphic objects (Leaves of the tree are primitive Graphic objects which know how to draw themselves, usually after looking up the values of various properties).

As per claims 19 and 20, they are the system versions of claims 9 and 10 respectively, and are rejected for the same reasons set forth in connection with the rejection of claims 9 and 10 above.

***Response to Arguments***

8. Applicant's arguments filed 9/25/2006 have been fully considered but they are not persuasive.

Per claims 1, 11, 21, and 22:

The Applicant states that: Main does not disclose accessing an option value not stored in the defined fields in the object using expressions of the same syntactic form.

In response, target_ptr->data is equivalent of (*target_ptr).data. Therefore, it is considered that addend.many_nodes and target_ptr->data are accessible by using the same syntactic form. The determining step is considered to be nothing more than compiler data type checking. If applicant means anything more, this must be brought out in the claims to further clarify the invention.

*Note: per applicant's request, references that prove that the book titled "Data Structures and other Objects Using C++" by Main and Savitch was available for public on 1997 are provided with this office action.

***Conclusion***

1. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a). A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and

any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

10. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Insun Kang whose telephone number is 571-272-3724. The examiner can normally be reached on M-R 6:30-5 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, MENG Al AN can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

I. Kang
Examiner

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 21